

#2

Docket No.: K-105

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Sunshin AN, Dongjin HAN, Wenzhe CUI,
Youngeun PARK, Shinhyuk KANG,
and Soo Hyun PARK

Serial No.: To be assigned

Filed: September 14, 1999

For: NETWORK MANAGEMENT SYSTEM AND METHOD

TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Assistant Commissioner of Patents
Washington, D. C. 20231

Sir:

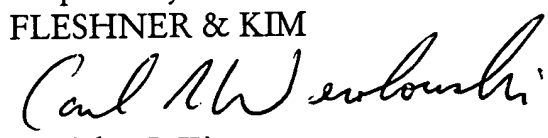
At the time the above application was filed, priority was claimed based on the

following application:

Korean Patent Application No. 43778/1998, filed October 17, 1998

A copy of each priority application listed above is enclosed.

Respectfully submitted,
FLESHNER & KIM



Daniel Y.J. Kim
Registration No. 36,186
Carl R. Wesolowski
Registration No. 40,372

P. O. Box 221200
Chantilly, Virginia 20153-1200
703 502-9440

Date: September 14, 1999
DYK/CRW:jld

JC575 U.S. PTO
09/395207
09/14/99

JC575 U.S. PTO
09/395207
09/14/99

대한민국 특허청
KOREAN INDUSTRIAL
PROPERTY OFFICE

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto
is a true copy from the records of the Korean Industrial
Property Office.

출원번호 : 1998년 특허출원 제43778호
Application Number

출원년월일 : 1998년 10월 17일
Date of Application

출원인 : 엘지정보통신 주식회사
Applicant(s)

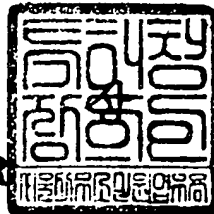
CERTIFIED COPY OF
PRIORITY DOCUMENT



1999년 7월 27일

특허청

COMMISSIONER



특허출원서

【출원번호】 98-043778

【출원일자】 1998/10/17

【발명의 국문명칭】 망 관리 시스템 및 클래스 동적 추가 방법

【발명의 영문명칭】 Network Management System And Class Dynamic Loading Method

【출원인】

【국문명칭】 엘지정보통신 주식회사

【영문명칭】 LG Information & Communications, Ltd.

【대표자】 서평원

【출원인코드】 11007112

【출원인구분】 국내상법상법인

【전화번호】 0343-50-7045

【우편번호】 150-010

【주소】 서울특별시 영등포구 여의도동 20번지

【국적】 KR

【대리인】

【성명】 김영철

【대리인코드】 A236

【전화번호】 02-725-4774

【우편번호】 110-727

【주소】 서울특별시 종로구 수송동 80-6 석탄회관빌딩 10층

【발명자】

【국문성명】 안순신

【영문성명】 AHN, Soon Shin

【주민등록번호】 500219-1056012

【우편번호】 158-050

【주소】 서울특별시 양천구 목동 목동아파트 1011동 201호

【국적】 KR

【발명자】

【국문성명】 한동진

【영문성명】 HAN, Dong Jin

【주민등록번호】 691204-1018311

【우편번호】 131-220

【주소】 서울특별시 중랑구 상봉동 신내엘지쌍용아파트 1308동 1203호

【국적】 KR

【발명자】

【국문성명】 최문철

【영문성명】 CHOI, Moon Chul

【주소】 서울특별시 성북구 안암동 5가 1번지

【국적】 CN

【발명자】

【국문성명】 박영은
【영문성명】 PARK, Young Eun
【주민등록번호】 710319-1120837
【우편번호】 136-073
【주소】 서울특별시 성북구 안암동3가 133-50
【국적】 KR

【발명자】

【국문성명】 강신혁
【영문성명】 KANG, Shin Hyuk
【주민등록번호】 721108-1953811
【우편번호】 136-075
【주소】 서울특별시 성북구 안암동5가 12-19
【국적】 KR

【발명자】

【국문성명】 박수현
【영문성명】 PARK, Soo Hyun
【주민등록번호】 640819-1046313
【우편번호】 132-020
【주소】 서울특별시 도봉구 방학동 507-1 우성아파트 201동 304호
【국적】 KR

【취지】 특허법 제42조의 규정에 의하여 위와 같이 출원합니다.

대리인 김영철 (인)

【심사청구】 특허법 제60조의 규정에 의하여 위와 같이 출원심사를 청구합니다.

대리인 김영철 (인)

【수신처】 특허청장 귀하

【수수료】

【기본출원료】	19 면	29,000 원
【가산출원료】	0 면	0 원
【우선권주장료】	0 건	0 원
【심사청구료】	5 항	269,000 원
【합계】	298,000 원	

【첨부서류】 1. 요약서, 명세서(및 도면) 각 1통

2. 출원서 부분, 요약서, 명세서(및 도면)을 포함하는 F0부분 1통

3. 위임장(및 동 번역문)

【요약서】

【요약】

본 발명은 망 관리 시스템 및 클래스 동적 추가 방법에 관한 것으로, 특히 망 관리 시스템의 실행 과정 중에도 DCL(Dynamic Class Loading) 기법을 이용하여 새로운 클래스의 동적 추가를 지원하도록 한 망 관리 시스템 및 클래스 동적 추가 방법에 관한 것이다.

DCL 루틴을 통해 관리 객체에의 동적 추가 기능을 지원하는 망 관리 시스템 코어와; 동작 실행 과정 중에 상기 망 관리 시스템 코어의 메모리에 상속되어 관리 객체를 생성하는 MODL(Managed Object Dynamic Library) 형식의 관리 객체 부분으로 분리되어 이루어진 망 관리 시스템에 의해 망 관리 시스템의 동작 연속성과 개발의 간편성을 보장해 준다.

【대표도】

도 3

【명세서】

【발명의 명칭】

망 관리 시스템 및 클래스 동적 추가 방법

【도면의 간단한 설명】

도 1은 일반적인 망 관리 시스템을 나타낸 구성 블록도.

도 2는 일반적인 망 관리 방법을 나타낸 플로우차트.

도 3은 본 발명의 실시예에 따른 망 관리 시스템을 나타낸 구성 블록도.

도 4는 본 발명의 실시예에 따른 망 관리 시스템에서 클래스 동적 추가 방법을 나타낸 플로우차트.

도 5는 도 4에 있어 DCL 루틴 수행 과정을 나타낸 플로우차트.

* 도면의 주요부분에 대한 부호의 설명 *

20 : 망 관리 시스템 코어

30 : MODL(Managed Object Dynamic Library)

21 : 통신 모듈

22 : 커널(Kernel)

23 : 포함 트리

24 : 관리 객체 프레임워크

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

본 발명은 망 관리 시스템 및 클래스 동적 추가 방법에 관한 것으로, 특히 망 관리 시스템의 실행 과정 중에도 DCL(Dynamic Class Loading) 기법을 이용하여 새로운

클래스의 동적 추가를 지원하도록 한 망 관리 시스템 및 클래스 동적 추가 방법에 관한 것이다.

일반적인 망 관리 시스템에서는 관리 객체를 정적으로 관리하는데, 해당 망 관리 시스템이 실행하기 전에 해당 관리 객체를 해당 망 관리 시스템 내의 다른 기능들과 같이 컴파일한 후에 해당 망 관리 시스템의 실행 프로그램 내에 저장시킨다.

이러한 망 관리 시스템의 구성은 도 1에 도시된 바와 같이, 통신 모듈(Communication Module; 11)과, 커널(Kernel; 12)과, 포함 트리(Containment Tree; 13)와, 관리 객체 프레임워크(Management Object Framework; 14)을 포함하여 이루어져 있다.

해당 통신 모듈(11)은 망 관리자와의 통신 기능을 담당하며, 해당 커널(12)은 망 관리자로부터 수신한 관리 동작 요청을 처리하며, 해당 포함 트리(13)는 관리 객체 간의 포함 관계를 구현한 구조체이며, 해당 관리 객체 프레임워크(14)은 관리 객체에 관련된 데이터와 코드를 제공한다.

해당 관리 객체 클래스는 망 관리 시스템이 생성될 때에 망 관리 시스템에 포함되는데, 해당 망 관리 시스템에서 생성된 관리 객체들은 해당 포함 트리(13)에 등록되고 해당 포함 트리(13)에 따라 관리 객체를 접근한다. 그리고, 클래스 정보 테이블은 망 관리 시스템 코어에 선언되어 있으며, 해당 관리 객체가 추가되기 전에는 해당 클래스 정보 테이블은 빈 상태이다.

상술한 바와 같이 구성된 망 관리 시스템은 도 2의 플로우차트와 같은 순서로 동작을 수행한다.

사용자가 기술한 GDMO(Guidelines for the Definition of Manage Objects) 문서 파일을 GDMO 컴파일러(Compiler)를 통하여 관리 객체 코드 및 해당 관리 객체 코드를 컴파일하기 위한 메이크파일(Makefile)을 생성하며(단계 T1), 해당 생성된 관리 객체 코드에 사용자가 관리 객체에 관한 코드인 사용자 코드를 추가한다(단계 T2).

그리고, 상기 GDMO 컴파일러를 통해 생성된 관리 객체 코드 및 상기 추가된 사용자 코드를 통신 모듈(11) 및 커널(12)과 같이 컴파일하여 실행 가능한 망 관리 시스템 프로그램을 생성한다(단계 T3).

이에, 상기 생성된 망 관리 시스템 프로그램을 실행시켜 상기 통신 모듈(11), 커널(12) 등의 초기화 과정을 수행하며(단계 T4), 그런 다음에 해당 망 관리 시스템을 대기 상태로 만들어 놓는데, 망 관리자로부터 관리 동작 요청을 수신할 때까지 해당 망 관리 시스템은 대기 상태를 계속 유지한다(단계 T5).

이 때, 망 관리자로부터 관리 동작 요청을 수신하면 해당 망 관리 시스템은 지정된 관리 객체에 대해 관리 동작을 수행하는데, 해당 관리 동작 요청이 관리 객체 생성인 경우를 확인하여(단계 T6) 클래스 정보 테이블에 저장된 관리 객체 생성 루틴을 수행하며(단계 T7) 관리 객체를 생성시켜 준다(단계 T8). 그런데, 만약 해당 관리 동작 요청이 관리 객체 생성이 아닌 다른 경우라면, 상기 포함 트리(13)를 통하여 선택된 관리 객체에 대하여 관리 동작을 수행한다(단계 T9).

이에 따라, 상기 관리 동작의 수행 결과를 망 관리자에게 통보한 후(단계 T10), 해당 망 관리 시스템은 다시 상기 제5단계(T5)의 대기 상태로 들어가는데, 이때 해당 망 관리 시스템이 필요없거나 해당 망 관리 시스템이 수정되어야 할 경우의 발생을

확인하여(단계 T11) 해당 망 관리의 수행을 중단시켜 준다(단계 T12).

이와 같이, 종래의 망 관리 시스템은 관리 객체들을 정적으로 관리하는 방법을 사용하고 있으므로, 해당 망 관리 시스템이 가동되기 전에 관리 객체들을 해당 망 관리 시스템이 생성될 때에 이미 포함하고 있어야 한다.

즉, 관리 객체 생성 요구가 들어올 경우에 이미 가지고 있는 관리 객체 생성 방법에 준하여 관리 객체를 생성하는 정적 관리 객체 관리 기법을 사용하는 망 관리 시스템은 새로운 관리 객체를 추가하거나 수정할 경우에 반드시 동작 수행을 중단시킨 후에 관리 객체를 다시 추가하거나 수정한 다음 다시 시스템을 작동해야 한다.

이와 같이, 종래에는 새로운 관리 객체를 추가하거나 수정할 경우에 반드시 망 관리 시스템의 동작 수행을 중단시켜야 하므로 전기 통신망 또는 지능망을 관리할 때 관리의 공백이 발생해 치명적일 수 있다.

【발명이 이루고자 하는 기술적 과제】

상술한 바와 같은 문제점을 해결하기 위해, 본 발명은 망 관리 시스템에 새로운 클래스의 동적 추가를 지원하는 DCL 방법을 이용한 관리 객체의 관리 및 사용 방법에 관한 것으로, 망 관리 시스템 코어와 관리 객체를 분리시켜 망 관리 시스템의 실행 과정 중에도 새로운 관리 객체를 동적으로 추가할 수 있도록 함으로써, 망 관리 시스템의 동작 연속성과 개발의 간편성을 보장하는 것을 목적으로 한다.

【발명의 구성 및 작용】

상기와 같은 목적을 달성하기 위한 본 발명의 실시예에 따른 망 관리 시스템은 DCL 루틴을 통해 신규 관리 객체의 동적 추가 기능을 지원하는 망 관리 시스템 코어와;

동작 실행 과정 중에 상기 망 관리 시스템 코어의 메모리에 상속되어 관리 객체를 생성하는 MODL 형식의 관리 객체 부분을 포함하여 이루어진 것을 특징으로 한다. 여기서, 상기 DCL 루틴을 지원하기 위해 동일한 베이스 클래스에서 상속된 관리 객체와 관리 객체 코드가 DLL(Dynamic Linking Library) 형식으로 컴파일된 DLL 파일에 관한 정보가 외부의 메타 데이터 파일에 저장되어 있는 것을 특징으로 한다.

또한, 상기와 같은 목적을 달성하기 위한 본 발명의 실시예에 따른 망 관리 시스템에서 클래스 동적 추가 방법은 GDMO 컴파일러를 통해 관리 객체 코드를 생성하고 해당 관리 객체 코드를 컴파일하기 위한 메이크파일과 메타 데이터 파일을 생성하는 과정과; 사용자 코드를 추가하여 해당 사용자 코드와 상기 관리 객체 코드를 컴파일시켜 MODL을 생성하는 과정과; 망 관리 시스템 코어를 초기화시켜 대기 상태를 유지하는 과정과; 클래스 동적 추가 지시의 수신을 확인하여 DCL 루틴을 수행하는 과정과; 상기 DCL 루틴의 수행 완료를 통보하고 상기 대기 상태로 되돌아가는 과정과; 상기 망 관리 시스템 코어가 필요없는 경우에 모든 동작 수행을 종료시키는 과정을 포함하여 이루어진 것을 특징으로 한다. 여기서, 상기 DCL 루틴을 수행하는 과정은 외부 기억 장치에 저장된 메타 데이터 파일에 의해 클래스 정보 테이블을 초기화시키는 단계와; 상기 메타 데이터 파일에서 기술된 관리 객체 이름 및 MODL 파일 이름에 의해 상기 MODL을 상기 망 관리 시스템 코어의 메모리에 상주시키는 단계와; 상기 MODL 내의 관리 객체 생성 루틴의 주소를 찾아 상기 클래스 정보 테이블에 등록시키는 단계와; 상기 클래스 정보 테이블에 등록한 관리 객체 생성 루틴 주소를 사용하여 관리 객체를 생성시키는 단계를 포함하여 이루어진 것을 특징

으로 한다.

이하 첨부된 도면을 참고하여 다음과 같이 설명한다.

도 3은 본 발명의 실시예에 따른 망 관리 시스템을 나타낸 구성 블록도이고, 도 4는 본 발명의 실시예에 따른 망 관리 시스템에서 클래스 동적 추가 방법을 나타낸 플로우차트이고, 도 5는 도 4에 있어 DCL 루틴 수행 과정을 나타낸 플로우차트이다.

본 발명의 실시예에 따른 망 관리 시스템을 도 3에 도시된 바와 같이, 독립적으로 실행될 수 있는 프로그램인 망 관리 시스템 코어(20)와, MODL 형식의 관리 객체 부분(30)으로 분리되어 이루어진다.

여기서, 상기 MODL(30)은 상기 망 관리 시스템 코어(20)가 동작 수행을 시작할 때에는 포함 관리하는 상기 망 관리 시스템 코어(20)에 포함되어 있지 않고 상기 망 관리 시스템 코어(20)가 동작 실행 과정 중에는 해당 망 관리 시스템 코어(20)에 추가된다. 또한, 상기 망 관리 시스템 코어(20)는 DCL 루틴(221)을 포함하고 있으며, 해당 DCL 루틴(221)을 지원하기 위하여 모든 관리 객체는 동일한 베이스 클래스에서 상속되며, 관리 객체 코드는 DLL 형식으로 컴파일되어 있으며, 관리 객체 및 DLL 파일에 관한 정보는 외부의 메타 데이터 파일에 저장된다.

그리고, 상기 망 관리 시스템 코어(20)는 상기 DCL 루틴(221)을 지원하기 위하여 망 관리자와의 통신 기능을 담당하는 통신 모듈(21)과, 망 관리자로부터 수신한 관리 동작 요청의 처리 기능을 수행하고 동시에 관리 객체를 추가하는 DCL 루틴(221)을 통해 관리 객체에의 동적 추가 기능을 지원하는 커널(22)과, 관리 객체 간의 포

함 관계를 구현한 구조체인 포함 트리(23)와, 실제 관리 객체를 포함하고 있지 않고 관리 객체의 동적 추가하는 기능을 지원하는 루틴 및 자료 구조만 가지고 있는 관리 객체 프레임웍(24)을 더 포함하여 이루어진다.

본 발명의 실시예에 따른 망 관리 시스템을 개발할 경우에 DCL 방법을 사용하므로써, 망 관리 시스템 코어(20)의 재 컴파일을 수행할 필요없이 동적 라이브러리(Dynamic Library) 형식의 MODL(30)만을 생성시켜 새로운 클래스의 수신 시에 해당 망 관리 시스템 코어(20)에 추가시켜 준다.

즉, 해당 망 관리 시스템 코어(20)의 실행 과정 중에도 관리 객체를 추가할 수 있으므로 해당 MODL(30)은 해당 망 관리 시스템 코어(20)의 실행 과정 중에도 개발될 수 있는데, 해당 새로운 관리 객체를 추가하기 위해서 해당 망 관리 시스템 코어(20)는 커널(22) 내에 구비되어 있는 DCL 루틴(221)을 사용하며, 해당 DCL 루틴(221)은 해당 망 관리 시스템 코어(20)가 관리 객체 추가의 지시를 수신한 후에 호출되어 실행되어진다.

그러면, 도 4의 플로우챗를 참고하여 상기 DCL 방법을 사용하는 망 관리 시스템의 개발 및 수행 과정을 살펴본다.

먼저, 사용자가 기술한 GDMO 문서 파일을 GDMO 컴파일러를 통하여 관리 객체 코드를 생성하게 되며, 해당 관리 객체 코드를 컴파일하기 위한 메이크파일과 메타 데이터 파일을 생성한다(과정 S1). 그리고, 사용자가 관리 객체에 관한 사용자 코드를 추가시켜 줌으로써, 해당 생성한 관리 객체 코드와 사용자 코드를 컴파일하여 상기 MODL(30)을 생성한다(과정 S2).

이에, 상기 망 관리 시스템 코어(20)에서는 망 관리 시스템 코어 프로그램을 실행시켜 통신 모듈(21), 커널(22) 등의 초기화 과정을 수행하여 대기 상태로 만들어 놓는데, 망 관리자로부터 관리 동작 요청 또는 관리 객체 추가 지시를 수신할 때까지 해당 망 관리 시스템 코어(20)는 대기 상태를 계속 유지한다(과정 S3).

이 때, 상기 망 관리 시스템 코어(20)가 망 관리자로부터 동적 추가 지시를 수신받으면(과정 S4), 상기 제1과정(S1)에서 생성시킨 메타 데이터 파일에 따른 클래스 정보 테이블의 초기화와 관리 객체의 추가 과정 순으로 수행하는 상기 DCL 루틴(221)을 수행한다(과정 S5).

그리고, 상기 DCL 루틴(221)의 수행 완료를 통보한 후에 상기 망 관리 시스템 코어(20)는 다시 상기 제3과정(S3)의 대기 상태로 되돌아간다(과정 S6).

한편, 상기 제4과정(S4)에서 망 관리자로부터 관리 동작 요청을 수신하였다면, 상기 망 관리 시스템 코어(20)는 종래의 기술과 동일한 방법으로 지정된 관리 객체에 대해 관리 동작을 수행한다. 즉, 해당 관리 동작 요청이 관리 객체 생성인 경우를 확인하여 클래스 정보 테이블에 저장된 관리 객체 생성 루틴을 수행하여 관리 객체를 생성시켜 주며, 만약 해당 관리 동작 요청이 관리 객체 생성이 아닌 다른 경우라면 포함 트리(23)를 통하여 선택된 관리 객체에 대하여 관리 동작을 수행하며, 이에 해당 관리 동작의 수행 결과를 망 관리자에게 통보한 후에 상기 망 관리 시스템 코어(20)는 다시 상기 제3과정(S3)의 대기 상태로 되돌아간다(과정 S7).

그런데, 이때 상기 망 관리 시스템 코어(20)가 필요없는 경우의 발생을 확인하면(과정 S8) 상기 망 관리 시스템의 모든 동작 수행을 중단시켜 준다(과정 S9).

또한, 도 5의 플로우챠트를 상기 제5과정(S5)의 DCL 루틴(221) 수행 과정을 조금 더 자세히 살펴보면, 외부 기억 장치에 저장된 메타 데이터 파일을 개방시켜 해당 메타 데이터 파일에 따라 클래스 정보 테이블을 초기화시켜 준다(단계 ST1).

그리고, 상기 메타 데이터 파일에서 기술된 관리 객체 이름 및 MODL 파일 이름을 판독하여 해당 판독한 이름에 따라 상기 제2과정(S2)에서 생성한 MODL(30)을 상기 망 관리 시스템 코어(20)의 메모리에 상주시킨다(단계 ST2).

이에, 외부 기억 장치에 저장시켜 두었던 MODL(30)을 상기 망 관리 시스템 코어(20)에서 실행시켜 상기 메타 데이터 파일에서 기술된 관리 객체 이름에 의하여 상기 MODL(30) 내의 관리 객체 생성 루틴의 주소를 찾아내며(단계 ST3), 해당 찾은 관리 객체 생성 루틴의 주소를 클래스 정보 테이블에 등록시켜 준다(단계 ST4).

이에 따라, 상기 클래스 정보 테이블에 저장된 관리 객체 생성 루틴 주소를 사용하여 관리 객체를 생성시켜 준다(단계 ST5).

【발명의 효과】

이상과 같이, 본 발명에 의해 망 관리 시스템 코어와 관리 객체를 분리시켜 DCL 방법을 이용하여 망 관리 시스템의 실행 과정 중에도 새로운 관리 객체를 동적으로 추가 또는 교체할 수 있어 망 관리 시스템의 동작 연속성과 개발의 간편성을 보장한다.

【특허청구범위】

【청구항 1】

DCL 루틴(221)을 통해 신규 관리 객체의 동적 추가 기능을 지원하는 망 관리 시스템 코어(20)와;

동작 실행 과정 중에 상기 망 관리 시스템 코어(20)의 메모리에 상속되어 관리 객체를 생성하는 MODL 형식의 관리 객체 부분(30)을 포함하여 이루어진 것을 특징으로 하는 망 관리 시스템.

【청구항 2】

제1항에 있어서,

상기 DCL 루틴(221)을 지원하기 위해 동일한 베이스 클래스에서 상속된 관리 객체와 관리 객체 코드가 DLL 형식으로 컴파일된 DLL 파일에 관한 정보가 외부의 메타데이터 파일에 저장되어 있는 것을 특징으로 하는 망 관리 시스템.

【청구항 3】

제1항에 있어서,

상기 망 관리 시스템 코어(20)는 상기 DCL 루틴(221)을 지원하기 위하여 망 관리자와의 통신 기능을 담당하는 통신 모듈(21)과;

상기 망 관리자로부터 수신한 관리 동작 요청의 처리 기능을 수행하고 상기 DCL 루틴(221)을 통해 신규 관리 객체의 동적 추가 기능을 지원하는 커널(22)과;

관리 객체 간의 포함 관계를 구현한 구조체인 포함 트리(23)와;

관리 객체의 동적 추가하는 기능을 지원하는 루틴 및 자료 구조만 가지고 있는 관

리 객체 프레임웍(24)을 포함하여 이루어진 것을 특징으로 하는 망 관리 시스템.

【청구항 4】

GDMO 컴파일러를 통해 관리 객체 코드를 생성하고 해당 관리 객체 코드를 컴파일하기 위한 메이크파일과 메타 데이터 파일을 생성하는 과정과;

사용자 코드를 추가하여 해당 사용자 코드와 상기 관리 객체 코드를 컴파일시켜 MODL을 생성하는 과정과;

망 관리 시스템 코어를 초기화시켜 대기 상태를 유지하는 과정과;

클래스 동적 추가 지시의 수신을 확인하여 DCL 루틴을 수행하는 과정과;

상기 DCL 루틴의 수행 완료를 통보하고 상기 대기 상태로 되돌아가는 과정과;

상기 망 관리 시스템 코어가 필요없는 경우에 모든 동작 수행을 종료시키는 과정을 포함하여 이루어진 것을 특징으로 하는 망 관리 시스템에서 클래스 동적 추가 방법.

【청구항 5】

제4항에 있어서,

상기 DCL 루틴을 수행하는 과정은 외부 기억 장치에 저장된 메타 데이터 파일에 의해 클래스 정보 테이블을 초기화시키는 단계와;

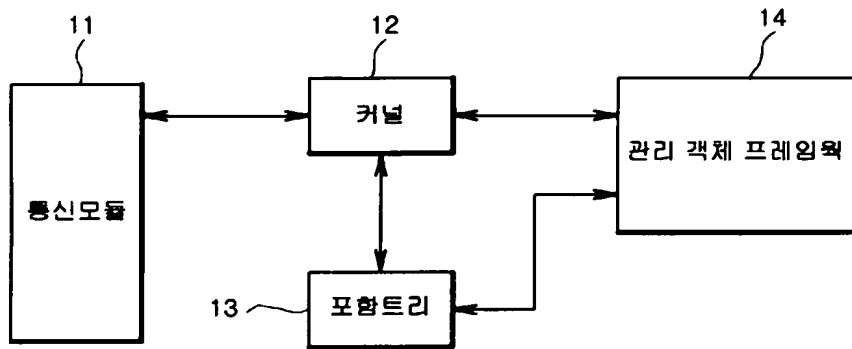
상기 메타 데이터 파일에서 기술된 관리 객체 이름 및 MODL 파일 이름에 의해 상기 MODL을 상기 망 관리 시스템 코어의 메모리에 상주시키는 단계와;

상기 MODL 내의 관리 객체 생성 루틴의 주소를 찾아 상기 클래스 정보 테이블에 등록시키는 단계와;

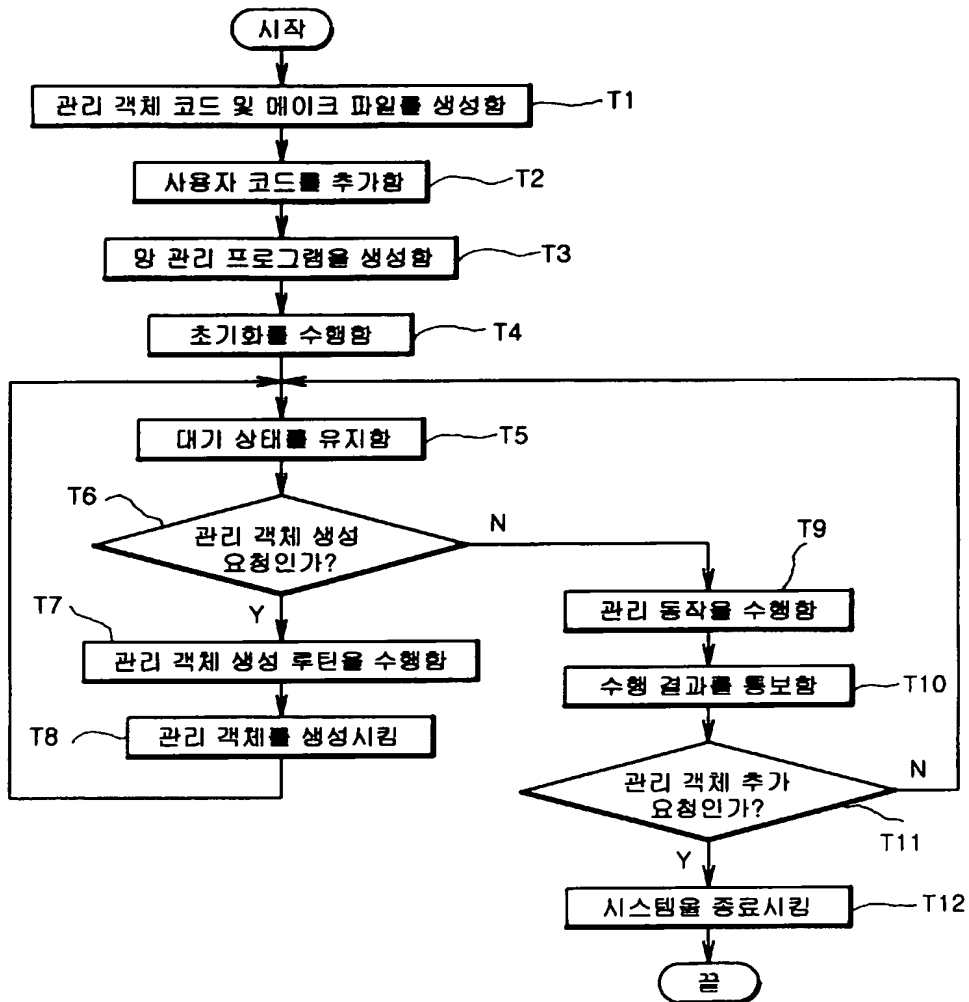
상기 클래스 정보 테이블에 등록된 관리 객체 생성 루틴 주소를 사용하여 관리 객체를 생성시키는 단계를 포함하여 이루어진 것을 특징으로 하는 망 관리 시스템에서 클래스 동적 추가 방법.

【도면】

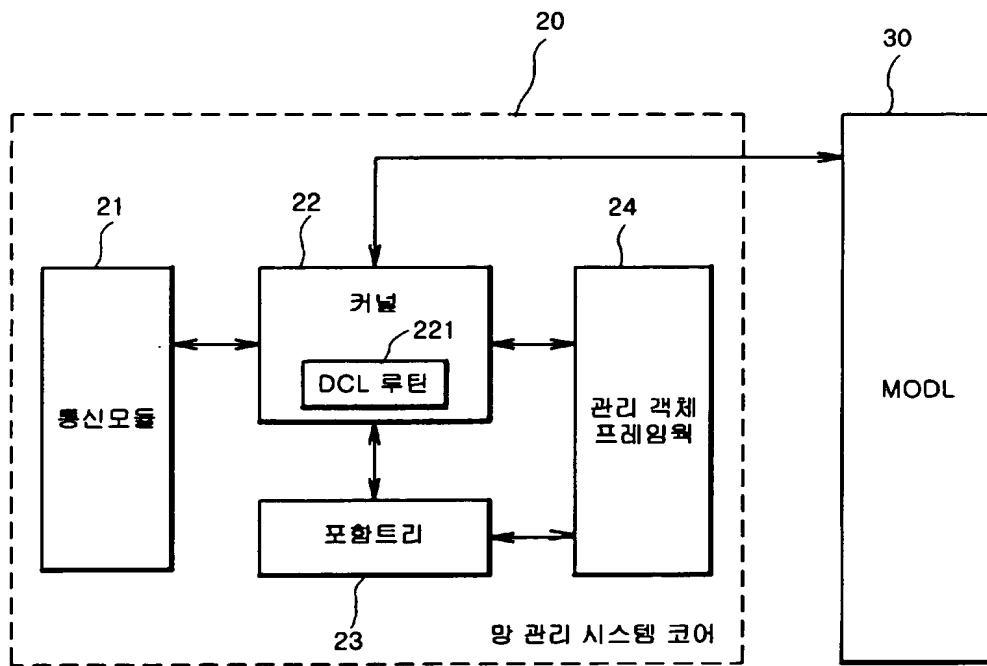
【도 1】



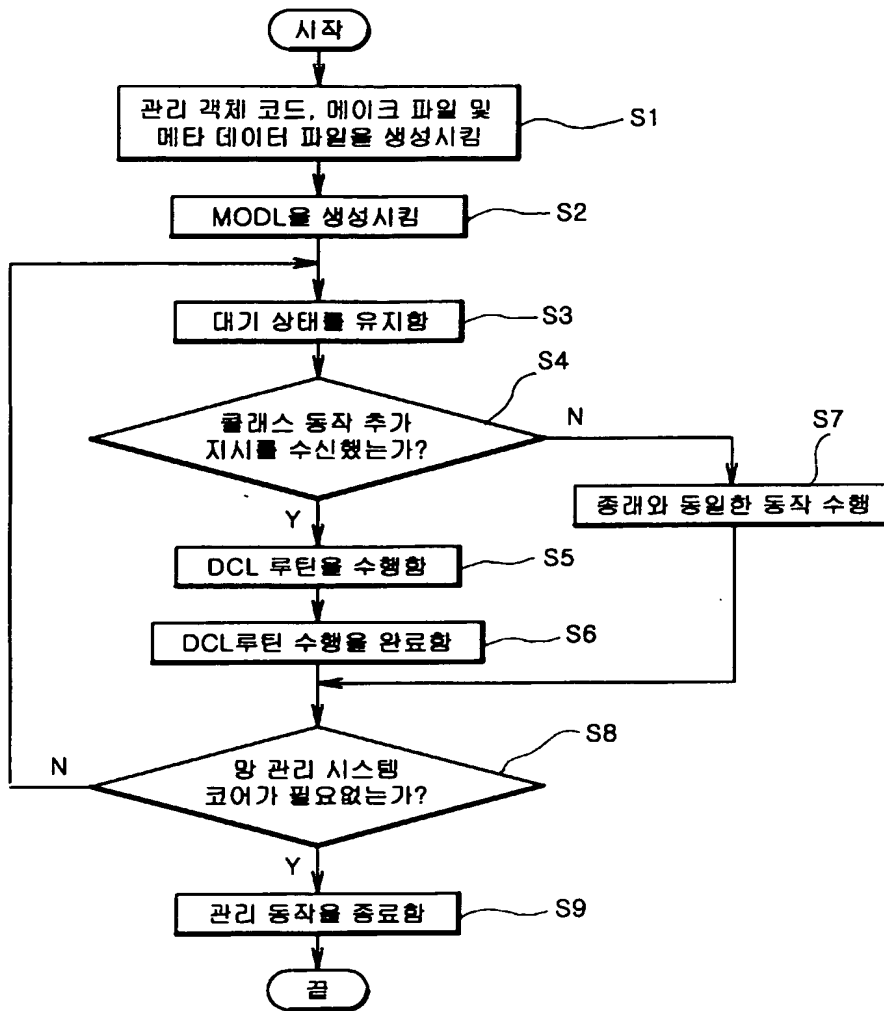
【도 2】



【도 3】



【도 4】



【도 5】

